

REMARKS

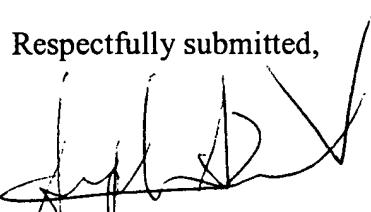
Claims 1-20 are pending in the application. Claims 20-43 have been added.

The claims and Specification have been amended to improve clarity. In addition, Figures 5A and 6 have been corrected and new formal drawings have been submitted. No new matter has been added.

In view of the amendments set forth herein, the application is believed to be in condition for allowance and a notice to that effect is solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the examiner is requested to telephone the undersigned.

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Commissioner for Patents, Washington, D.C. 20231, on <u>March 18, 2002</u> .	
<u>Attorney for Applicant(s)</u>	<u>31/8/02</u> Date of Signature

Respectfully submitted,


Stephen A. Terrile
Attorney for Applicant(s)
Reg. No. 32,946

APPENDIX A

The following is a "Marked Up" version of the entire set of claims showing the changes that the accompanying submission makes to the claims of Serial No. 10/067,040:

1. (As Filed) An apparatus for interfacing between a systems management system and a remote services system comprising:
 - a systems management application program interface;
 - a systems management integrator application program interface, the systems management integrator application program interface providing a normalization point where data from the systems management system is normalized to a remote services system standard; and
 - an integrator coupled between the systems management application program interface and the systems management integrator application program interface, the integrator collecting and detecting information from the systems management system.
2. (As Filed) The apparatus of claim 1 wherein the systems management integrator application program interface includes:
 - a forward calls component, the forward calls component providing forwards calls from the systems management system to the remote services system.
3. (As Filed) The apparatus of claim 2 wherein the systems management integrator application program interface includes:
 - a back-channel calls component, the back-channel calls component providing back-channel calls from the remote services system to the system management system.
4. (As Filed) The apparatus of claim 1 wherein the systems management integrator application program interface provides generic message interfaces as well as specific message interfaces, the specific message interfaces being provided to enable the remote services system to handle a message without having to inspect the contents of the message.

5. (As Filed) The apparatus of claim 4 wherein the specific message interfaces include an alarm message.

6. (As Filed) The apparatus of claim 4 wherein the specific message interfaces include an event message.

7. (As Filed) The apparatus of claim 4 wherein:
the generic message interfaces includes data, the data including a class, and
the integrator sets the class of the data.

8. (As Filed) The apparatus of claim 3 wherein the systems management integrator application program interface:

provides support for declaring capabilities of a support instance at registration time via
the forward channel, and
provides support for a request of the capabilities of the support instance via the back-
channel after registration time.

9. (As Filed) The apparatus of claim 1 wherein the systems management integrator application program interface provides an interface between the integrator and a remote services infrastructure.

10. (As Filed) The apparatus of claim 8 wherein the systems management integrator application program interface is coupled to a remote services proxy of the remote services infrastructure.

11. (As Filed) A systems management integrator application program interface for a remote services system comprising:

a forward calls component, the forward calls component providing forwards calls from a systems management system to the remote services system; and
a back-channel calls component, the back-channel calls component providing back-
channel calls from the remote services system to the system management system.

12. (As Filed) The systems management integrator application program interface of claim 11 wherein the systems management integrator application program interface provides generic message interfaces as well as specific message interfaces, the specific message interfaces being provided to enable the remote services system to handle a message without having to inspect the contents of the message.

13. (As Filed) The systems management integrator application program interface of claim 12 wherein the specific message interfaces include an alarm message.

14. (As Filed) The systems management integrator application program interface of claim 12 wherein the specific message interfaces include an event message.

15. (As Filed) The systems management integrator application program interface of claim 12 wherein

the generic message interfaces includes data, the data including a class, and an integrator sets the class of the data.

16. (As Filed) The systems management integrator application program interface of claim 11 wherein the systems management integrator application program interface provides support for declaring capabilities of a support instance at registration time via the forward channel, and provides support for a request of the capabilities of the support instance via the back-channel after registration time.

17. (As Filed) The systems management integrator application program interface of claim 11 wherein the systems management integrator application program interface provides an interface between a systems management platform and a remote services infrastructure.

18. (As Filed) The systems management integrator application program interface of claim 17 wherein the systems management integrator application program interface is coupled to a remote services proxy of the remote services infrastructure.

19. (As Filed) A method for interfacing between a systems management system and a remote services system comprising:

providing a systems management application program interface; collecting and detecting information from the systems management system using a systems management integrator, the systems management integrator interfacing with the systems management system via the systems management application program interface;

providing and receiving information to the remote services system, the providing and receiving information being via a systems management integrator application program interface, the systems management integrator application program interface providing a normalization point where data from the systems management system is normalized to a remote services system standard.

20. (New) A remote services proxy comprising:

a proxy foundation module, the proxy foundation module managing data normalization tasks;

a proxy integrator application program interface (API) module coupled to the proxy foundation module, the proxy API module providing a normalization point to the proxy foundation module;

a proxy identifier (ID) management module coupled to the proxy foundation module, the proxy ID management module managing allocation of unique identifiers for the remote services proxy and for any support instances registered through the remote services proxy; and

a proxy queuing module coupled to the proxy foundation module, the proxy queuing module queuing data for later transmission.

21. (New) The remote services proxy of claim 20 wherein the proxy foundation module and the proxy API module communicate using an Inter-Process Communication (IPC) implementation.

22. (New) The remote services proxy of claim 20 wherein

the proxy integrator API module is implemented using one of a plurality of different programming languages to correspond to a systems management software language.

23. (New) The remote services proxy of claim 20 wherein the proxy queuing module includes a plurality of features for managing queuing of data.

24. (New) The remote services proxy of claim 23 wherein the plurality of features include a priority feature and a time for data to live feature.

25. (New) The remote services proxy of claim 20 wherein the proxy ID management module manages mapping between a systems management ID and a remote services ID.

26. (New) The remote services proxy of claim 20 wherein the remote services ID is keyed off a unique customer ID.

27. (New) A remote services proxy comprising a proxy daemon, the proxy daemon providing infrastructure management services; and, at least one communication mechanism coupled between the proxy daemon and an integrator application program interface (API).

28. (New) The remote services proxy of claim 27 wherein the communication mechanism is an Inter-Process Communication (IPC) communication mechanism.

29. (New) The remote services proxy of claim 27 wherein the infrastructure management services include at least one of software upload services, software update services and proxy status services.

30. (New) The remote services proxy of claim 27 further comprising: a communications module coupled to the proxy daemon.

31. (New) The remote services proxy of claim 30 wherein the communications module includes a routing module.

32. (New) The remote services proxy of claim 30 wherein the communications module includes a queuing module.

33. (New) The remote services proxy of claim 30 wherein the communications module includes a communications/encryption module.

34. (New) A method for initializing a remote services proxy in a remote services system comprising

consulting a proxy configuration file to determine whether the remote services proxy has already registered with the remote services system;

sending a registration message to the remote services system if the remote services proxy has not yet registered;

awaiting a positive acknowledgement of registration before completing initialization and beginning full operation; and

storing the positive acknowledgement in persistent data of the remote services proxy.

35. (New) The method for initializing a remote services proxy of claim 34 further comprising:

connecting an integrator to the remote services proxy via a remote services integrator application program interface (API); and,

determining whether the integrator has previously registered with the remote services system.

36. (New) The method for initializing a remote services proxy of claim 35 wherein the determining whether the integration has previously registered is based upon a remote services ID of the integrator.

37. (New) The method for initializing a remote services proxy of claim 35 further comprising:

allocating a remote services ID for the integrator by the remote services proxy when the integrator has not previously registered with the remote services system; and sending the remote services ID to the remote services system in a registration message.

38. (New) The method for initializing a remote services proxy of claim 37 further comprising

sending additional information to the remote services system in the registration message, the additional information including at least one of a module ID for the integrator and version information for the integrator.

39. (New) The method for initializing a remote services proxy of claim 35 further comprising:

registering support instances being managed by the integrator.

40. (New) The method for initializing a remote services proxy of claim 39 wherein registering the support instances being managed by the integrator enables the remote services system to perform entitlement checking against the instance and the services being provided to a customer and enables the remote services system to send data to a particular support instance to provide a particular service action.

41. (New) A remote services proxy comprising:

a proxy foundation module, the proxy foundation module managing data normalization tasks; and

a proxy integrator application program interface (API) module coupled to the proxy foundation module, the proxy API module providing a normalization point to the proxy foundation module.

42. (New) The remote services proxy of claim 40 wherein the proxy ID management module manages mapping between a systems management ID and a remote services ID.

43. (New) The remote services proxy of claim 40 wherein

the remote services ID is keyed off a unique customer ID.

APPENDIX B

"Marked Up" paragraphs showing the changes that the accompanying submission makes to the specification of Serial No. 10/067,040:

Please replace the paragraph beginning on Page 1, line 11 with the following amended paragraph:

This application relates to co-pending United States patent application Serial No. [] 10/067,074, attorney docket number P7225, filed on February 4, 2002, entitled "Remote Services Message System to Support Redundancy of Data Flow" and naming Michael J. Wookey, Trevor Watson and Jean Chouanard as inventors, the application being incorporated herein by reference in its entirety.

Please replace the paragraph beginning on Page 1, line 17 with the following amended paragraph:

This application relates to co-pending United States patent application Serial No. [] 10/066,950, attorney docket number P7229, filed on February 4, 2002, entitled "Remote Services Delivery Architecture" and naming Michael J. Wookey, Trevor Watson and Jean Chouanard as inventors, the application being incorporated herein by reference in its entirety.

Please replace the paragraph beginning on Page 1, line 24 with the following amended paragraph:

This application relates to co-pending United States patent application Serial No. [] 10/066,828, attorney docket number P7230, filed on February 4, 2002, entitled "Prioritization of Remote Services Messages Within a Low Bandwidth Environment" and naming Michael J. Wookey, Trevor Watson and Jean Chouanard as inventors, the application being incorporated herein by reference in its entirety.

Please replace the paragraph beginning on Page 1, line 30 with the following amended paragraph:

This application relates to co-pending United States patent application Serial No.

[] 10/067,165, attorney docket number P7231, filed on February 4, 2002, entitled "Remote Services System Back-Channel Multicasting" and naming Michael J. Wookey, Trevor Watson and Jean Chouanard as inventors, the application being incorporated herein by reference in its entirety.

Please replace the paragraph beginning on Page 2, line 4 with the following amended paragraph:

This application relates to co-pending United States patent application Serial No.

[] 10/066,841, attorney docket number P7233, filed on February 4, 2002, entitled "Remote Services System Data Delivery Mechanism" and naming Michael J. Wookey, Trevor Watson and Jean Chouanard as inventors, the application being incorporated herein by reference in its entirety.

Please replace the paragraph beginning on Page 2, line 10 with the following amended paragraph:

This application relates to co-pending United States patent application Serial No.

[] 10/066,914, attorney docket number P7234, filed on February 4, 2002, entitled "Remote Services WAN Connection Identity Anti-spoofing Control" and naming Michael J. Wookey, Trevor Watson and Jean Chouanard as inventors, the application being incorporated herein by reference in its entirety.

Please replace the paragraph beginning on Page 2, line 16 with the following amended paragraph:

This application relates to co-pending United States patent application Serial No.

[] 10/066,075, attorney docket number P7235, filed on February 4, 2002, entitled "Automatic Communication Security Reconfiguration for Remote Services" and naming Michael J. Wookey, Trevor Watson and Jean Chouanard as inventors, the application being incorporated herein by reference in its entirety.

Please replace the paragraph beginning on Page 34, line 13 with the following amended paragraph:

Referring again to Figure [M]14, the remote services proxy 210 enables multiple integrators 212 running on the same host to connect through a shared service layer to the remote services system 100. The remote services proxy 210 also provides a means by which requests from the remote services system 100 to the systems management platform [N]1506 can be received and routed correctly. The proxy 210 is fast and lightweight by running in native code on the host.

Please replace the paragraph beginning on Page 35, line 4 with the following amended paragraph:

In session mode (i.e., there is a forward and back-channel for messages), the remote services proxy daemon [M]1414 expects to get a positive acknowledgement of registration before the proxy daemon [M]1414 begins full operation. Receipt of positive acknowledgement is stored in persistent data of the remote services proxy 210. Where there is no back-channel capability, however, (i.e., the system is in message mode) the remote services proxy 210 determines whether session or message mode is active through the communications layer API 440.

Please replace the paragraph beginning on Page 36, line 17, with the following amended paragraph:

The next stage of the registration process is for the integrator 212 to register all support instances that the integrator 212 is managing. A support instance is a device, host or software component which is being managed by the systems management platform [N]1506 to which the integrator 212 is connected. Registration of support instances allows the remote services system 100 to perform entitlement checking against the instance and the services being provided to the customer and enables the remote services system to send data or instructions to that particular support instance to provide a particular service action.

Please replace the paragraph beginning on Page 37, line 13, with the following amended paragraph:

Support instance registration occurs dynamically during the lifetime of the integrator's 212 connection to the system management platform [N]1506. For example, when a new agent (i.e., support instance) is added to the system management topology, the system management platform notifies the integrator 212 which then sends a registration request for that support instance. The integrator 212 only registers support instances which have an agent installed.

Please replace the paragraph beginning on Page 38, line 25, with the following amended paragraph:

Referring again to Figure 14, the remote services proxy 210 uses queuing module [M]1432 to provide persistent queuing of requests to be sent to the remote services system 100. Accordingly, in the event of a temporary network outage, or the failure of a local or remote MLM, data is not lost.

Please replace the paragraph beginning on Page 41, line 30, with the following amended paragraph:

For availability purposes, the proxy 210 sends a status heartbeat back to the remote services system 100 at regular periods. The period depends on the deployment model and the communications module in use. The period is configurable. Where the communications module [M]1428 allows for back-channel communications, the proxy 210 may receive a back-channel request when sending out the status heartbeat message. The proxy 210 makes a regular callback on the back-channel of the integrator API 430 to each of the integrators 212 which have registered with the proxy 210. This callback requests the status of the integrator 212, the status of the system management platform and optionally the status of each support instance. Once the status has been gathered for all active integrators 212, the proxy 210 adds its own status and sends the entire status as a message back to the remote services system 100.

Please replace the paragraph beginning on page 43, line 6, with the following amended paragraph:

Because the integrator 212 may be a component of the systems management platform [N]1506, it may be difficult to update this integrator automatically unless provided for by the systems management vendor. Each integration module includes a capability which determines whether or not the integration module can be updated automatically. If this capability is defined,

this functionality is provided for in the integration module's API. The integration module itself then receives the notification of the update via the API and is responsible for locating, installing and starting the update. When an integration module cannot be updated automatically, the customer is notified of the update via an administration portal and is provided instructions (or a script) to perform the update manually.

Please replace the paragraph beginning on page 43, line 16, with the following amended paragraph:

Not all systems management platforms [N]1506 support loading of modules into an agent layer, and even those that do may not support the loading programmatically. The systems management platforms [N]1506 that do support programmatic loading of modules provide an implementation for the appropriate API call in the integrator API 430. The proxy 210 may then call this API when a new module is to be loaded. To save passing large volumes of data through the API, a file name (or URL) may be passed to the integrator 212. The integrator 212 is then responsible for loading and processing the update. Where the systems management platform [N]1506 does not support programmatic loading of modules, the customer is advised of a new module (or update) via the administration portal and is provided instructions (or a script) via which the module can be manually added.

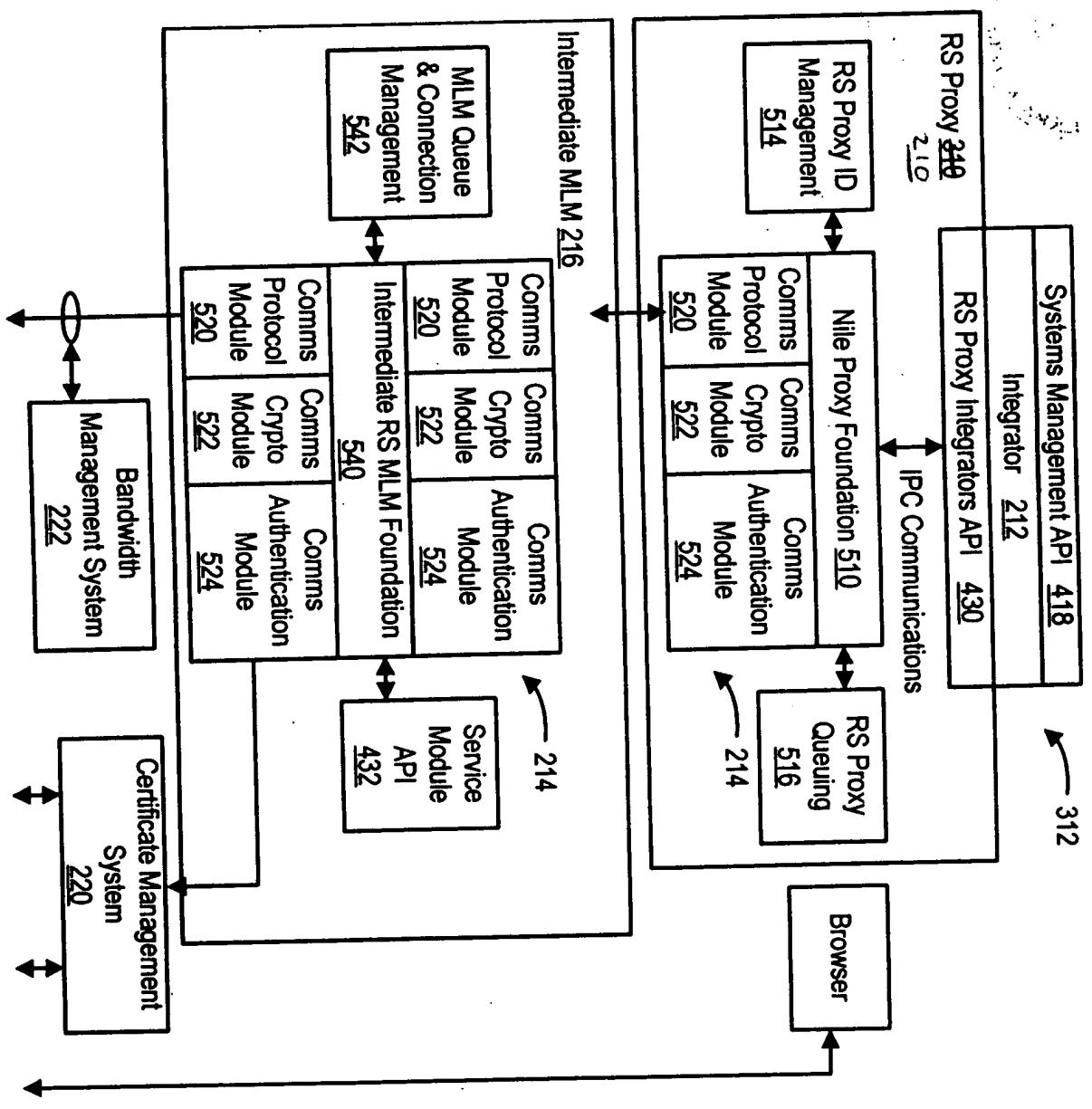


FIG. 5A

Figure 5
Legend

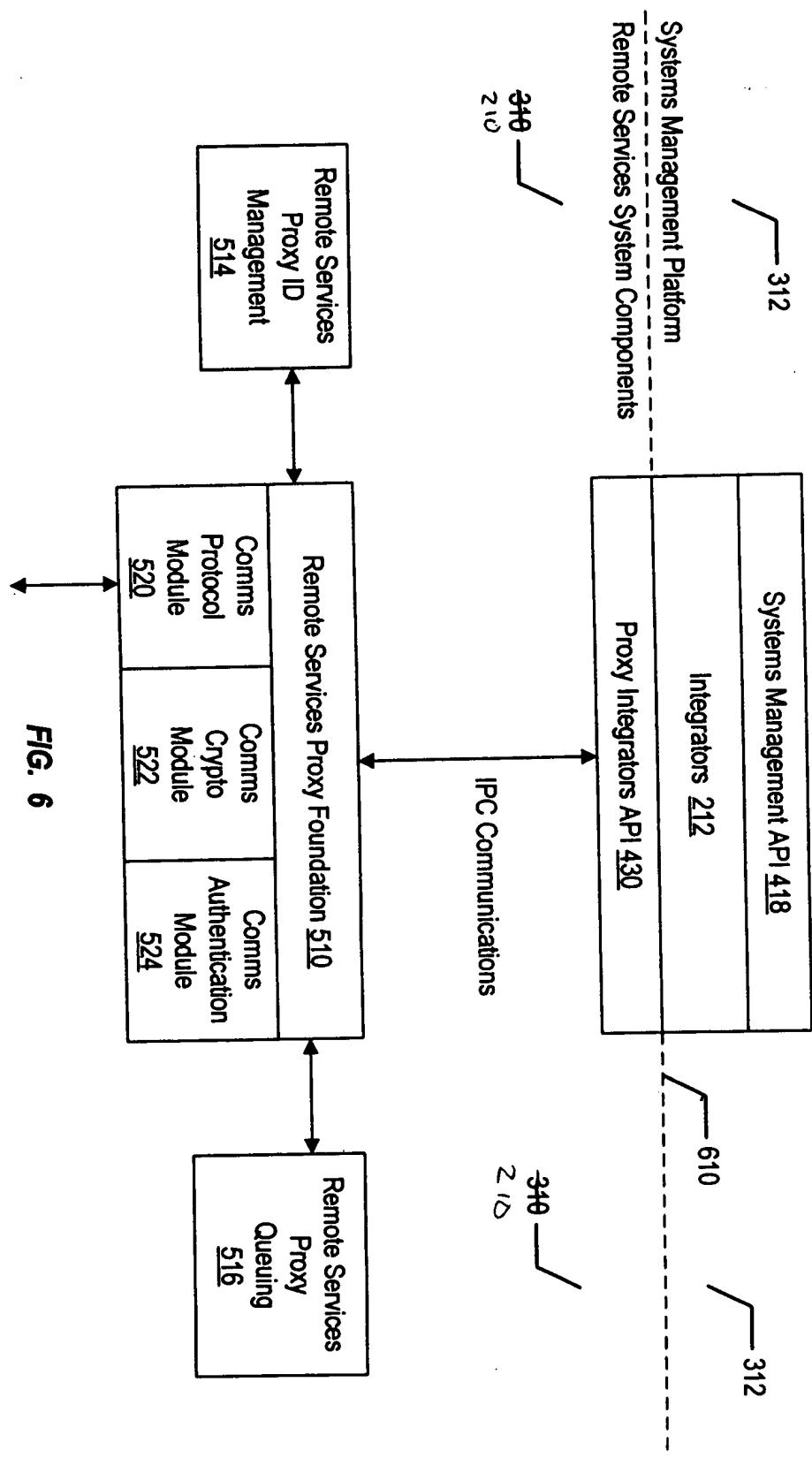


FIG. 6